Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1.   (currently amended) In a computer system, a method for providing improved real time command execution in a non real time operating system, comprising:

     executing at least one application at a user mode level of said computer system;

     having said at least one application at said user mode level determine a sequence to be followed for a set of ~~asynchronous~~ commands;

     providing from said at least one application said sequence of ~~asynchronous~~ commands to a privileged mode of said computer system to be executed in real time;

     storing said sequence of ~~asynchronous~~ commands in a command queue to be accessible from a privileged mode level of said comput<u>er</u>~~ing~~ system; and

     ~~implementing~~ <u>executing</u> one at a time each of <u>said commands from</u> said stored <u>sequence of</u> ~~asynchronous~~ commands.

2.   (currently amended) The method as claimed in claim 1, wherein a plurality of sequences of asynchronous commands is provided, each sequence being related to a corresponding application thread, further wherein said storing ~~of~~ <u>said</u> sequence of ~~asynchronous~~ commands is

performed in a corresponding queue from the execution of said corresponding application thread.

3.   (currently amended) The method as claimed in claim 1, wherein a synchronous command is added to said sequence of ~~asynchronous~~ commands, said at least one application sleeping until said synchronous command is executed.

4.   (currently amended) The method as claimed in claim 2, wherein a synchronous command is added to said sequence of asynchronous commands, said corresponding application thread sleeping until said synchronous command is executed.

5.   (currently amended) The method as claimed in claim 1, wherein said non real time operating system is Microsoft Windows and said ~~step of~~ storing said sequence of commands is performed through execution of a driver routine from a DLL file.

6.   (currently amended) The method as claimed in claim 5, wherein said ~~step of~~ providing said sequence of commands involves said commands being pushed one at a time ~~into said sequence~~ through a system call.

7.   (currently amended) The method as claimed in claim 1, wherein at least one of said stored commands is a branch command to control the order of execution of said stored commands.

8.   (currently amended) The method as claimed in claim 1, wherein said ~~step of implementing~~executing said commands from said stored sequence of commands is done at a different privileged mode level.

9.   (currently amended) The method as claimed in claim 8, wherein said different privileged mode level is that of ~~the~~ Interrupt Service Routine,

3

whereby ~~the~~ delay between the execution of successive commands is minimized.

10. (currently amended) The method as claimed in claim 9, wherein said non real_-time operating system is Microsoft Windows.

11. (currently amended) The method as claimed in claim 1, wherein said sequence of commands process ~~the~~ a same data set.

12. (currently amended) The method as claimed in claim 11, wherein said same data set is a video camera image being captured and processed in real_-time.

13. (currently amended) The method as claimed in claim 1, wherein said ~~step of~~ providing said sequence of commands involves said commands being pushed one at a time ~~into said sequence~~ through a system call.

14. (currently amended) The method as claimed in claim 1, wherein said ~~step of~~ storing said sequence of commands is performed through execution of a driver routine from a system file.